Advice for Applying Machine Learning

전 재 욱

Embedded System 연구실
성균관대학교

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Debugging A Learning Algorithm

- Regularized linear regression to predict housing prices

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n}\theta_j^2\right]$$

- When we test our hypothesis on a new set of houses,
  - we may find that it makes unacceptably large errors in its predictions.
    - What should we try next?
      - Get more training examples
      - Try smaller sets of features
      - Try getting additional features
      - Try adding polynomial features $(x_1^2,\ x_2^2,\ x_1x_2, \cdots)$
      - Try decreasing λ
      - Try increasing λ

Debugging A Learning Algorithm

- Get more training examples
  - Sometimes more data do not help
    - Often they does though,
      - although we should always do some preliminary testing to make sure more data will actually make a difference
- Try smaller sets of features
  - Carefully select small subset
  - We can do this by hand, or use some dimensionality reduction technique (e.g. PCA)
- Try getting additional features
  - Sometimes this is not helpful
  - We need to look at the data
  - This can be very time consuming

Debugging A Learning Algorithm

- Try adding polynomial features $(x_1^2,\ x_2^2,\ x_1x_2, \cdots)$
  - …
- Building our own, new, better features based on our knowledge of the problem
  - Can be risky if we accidentally over fit our data by creating new features which are inherently specific/relevant to our training data
- Try decreasing λ or increasing λ
  - Change how important the regularization term is in our calculations

Debugging A Learning Algorithm

- These changes can become major projects
  - 6 months more

  - Most common method for choosing one of these examples is to go by gut feeling (randomly)
  - Many times, we may spend huge amounts of time only to discover that the avenue is fruitless

- Simple techniques to rule out half the things on the list
  - We can save our time a lot.

Debugging A Learning Algorithm

- Machine learning diagnostic
  - Diagnostic:
    - A test that we can run

      to gain insight what is/(is NOT) working with a learning algorithm, and gain guidance as to how best to improve its performance.

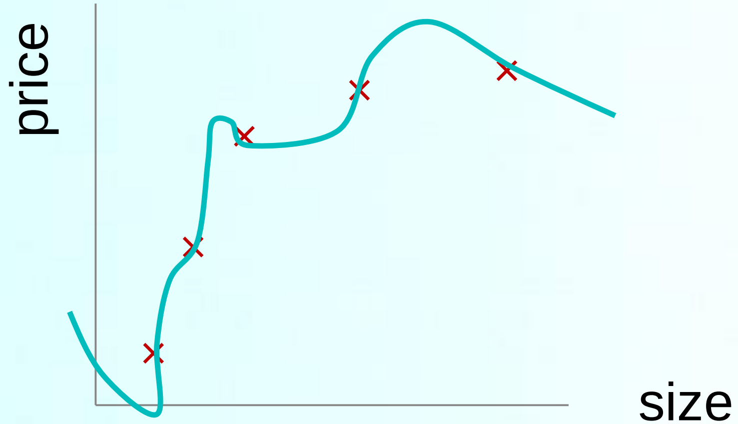
- Diagnostics can take time to implement (maybe week),
  - but doing so can be a very good use of our time.

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Evaluating A Hypothesis

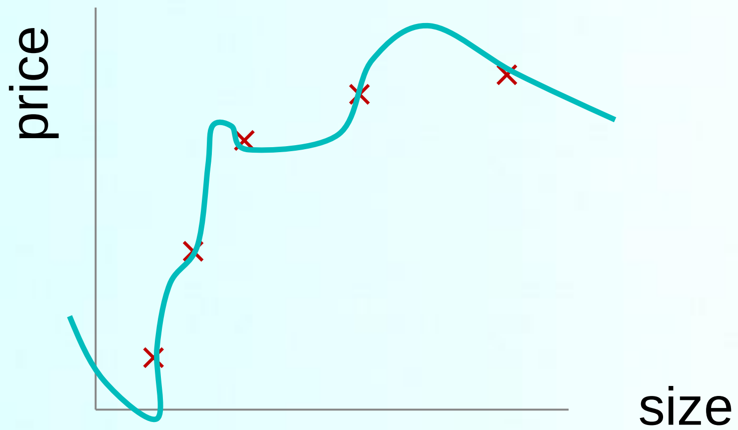
Evaluating A Hypothesis



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

- Fails to generalize to new examples not in training set
  - Low errors, but overfit (left Fig.)

Evaluating A Hypothesis



- Fails to generalize to new examples not in training set
  - Low errors, but overfit (left Fig.)
- Is a hypothesis overfitting?
  - Could plot $h_\theta(x)$
    - But with lots of features, it may be impossible to plot

$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

x_1: size of house
x_2: # of bedrooms
x_3: # of floors
x_4: age of house
x_5: average income in nbd
x_6: kitchen size
⋮
x_{100}

Evaluating A Hypothesis

- Dataset

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

- Standard way to evaluate a hypothesis
  - Split data into two portions
    - 1^{st}: training set
    - 2^{nd}: test set
  - Typical split
    - 70:30 (training : test)
- If data are ordered, send a random percentage
  - (Or randomly order, then send data)
  - Data are typically ordered in some way anyway

Evaluating A Hypothesis

Dataset

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

Training set
70%

$$\left(x^{(1)}, y^{(1)}\right)$$
$$\left(x^{(2)}, y^{(2)}\right)$$
$$\vdots$$
$$\vdots$$
$$\left(x^{(m)}, y^{(m)}\right)$$

Test set
30%

$$\left(x_{test}^{(1)}, y_{test}^{(1)}\right)$$
$$\left(x_{test}^{(2)}, y_{test}^{(2)}\right)$$
$$\vdots$$
$$\left(x_{test}^{(m_{test})}, y_{test}^{(m_{test})}\right)$$

Training/Testing Procedure

- Training/testing procedure for linear regression
  - Learn parameter θ from training data (Min training error $J(\theta)$)
    - 70% of total data
  - Compute test set error:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta \left(x_{test}^{(i)} \right) - y_{test}^{(i)} \right)^2$$

Training/Testing Procedure

- Training/testing procedure for logistic regression
  - Learn parameter θ from training data (Min training error $J(\theta)$)
    - 70% of total data
  - Compute test set error:

$$J_{test}(\theta) = -\frac{1}{m_{test}}\left[\sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_\theta\left(x_{test}^{(i)}\right) + \left(1 - y_{test}^{(i)}\right) \log(1 - h_\theta\,(x_{test}^{(i)}))\right]$$

  - Or compute test error
    - Test error = $\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err\left(h_\theta\left(x_{test}^{(i)}\right), y_{test}^{(i)}\right)$

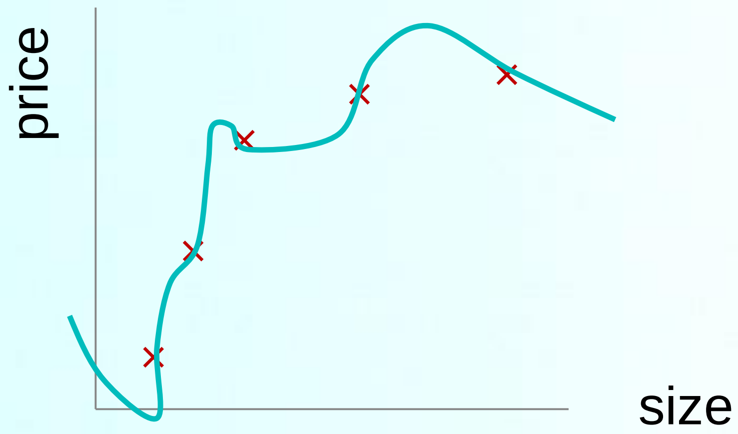
where misclassification error (0/1 misclassification error)

➢ $err(h_\theta(x), y) = \begin{cases} 1 & if\ h_\theta(x) \geq 0.5,\ \ y = 0 \\ & or\ if\ h_\theta(x) < 0.5,\ \ y = 1 \\ 0 & otherwise \end{cases}$

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Overfitting Example



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

- Once parameters $\Theta_0, \Theta_1, \cdots, \Theta_4$ were fit to some set of data (training set),
  - the error of the parameters as measured on that data (the training error $J(\Theta)$)
    - is likely to be lower than
      - the actual generalization error.

Model Selection

- How to chose regularization parameter or
  degree of polynomial (model selection problems)?

- Model selection problem
  - Try to choose the degree for a polynomial to fit data
    - $d = 1$ $h_\theta(x) = \theta_0 + \theta_1 x$
    - $d = 2$ $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$
    - $d = 3$ $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$
    - $\vdots$
    - $d = 10$ $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$

Model Selection

- How to chose regularization parameter or degree of polynomial (model selection problems)?

- Model selection problem
  - Try to choose the degree for a polynomial to fit data
    - $d = 1$ $h_\theta(x) = \theta_0 + \theta_1 x$ $\xrightarrow[\min_{\Theta} J(\Theta)]{\text{Min a training error}}$ $\Theta^{(1)}$
    - $d = 2$ $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$
    - $d = 3$ $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$
    - $\vdots$
    - $d = 10$ $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$

Model Selection

- How to chose regularization parameter or degree of polynomial (model selection problems)?

- Model selection problem
  - Try to choose the degree for a polynomial to fit data
    - $d = 1 \quad h_\theta(x) = \theta_0 + \theta_1 x \longrightarrow \Theta^{(1)}$
    - $d = 2 \quad h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \xrightarrow{\min_{\boldsymbol{\Theta}} \boldsymbol{J}(\boldsymbol{\Theta})} \Theta^{(2)}$
    - $d = 3 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 \longrightarrow \Theta^{(3)}$
    - $\vdots \qquad \vdots$
    - $d = 10 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \longrightarrow \Theta^{(10)}$

Model Selection

- How to chose regularization parameter or degree of polynomial (model selection problems)?

- Model selection problem
  - Try to choose the degree for a polynomial to fit data
    - $d = 1 \quad h_\theta(x) = \theta_0 + \theta_1 x \longrightarrow \Theta^{(1)} \rightarrow J_{test}(\Theta^{(1)})$
    - $d = 2 \quad h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \longrightarrow \Theta^{(2)} \rightarrow J_{test}(\Theta^{(2)})$
    - $d = 3 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 \longrightarrow \Theta^{(3)} \rightarrow J_{test}(\Theta^{(3)})$
    - $\vdots \qquad \vdots \qquad \vdots$
    - $d = 10 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \longrightarrow \Theta^{(10)} \rightarrow J_{test}(\Theta^{(10)})$

$\min_{\Theta} J(\Theta)$

Test set error

Model Selection

- How to chose regularization parameter or degree of polynomial (model selection problems)?

- Model selection problem
  - Try to choose the degree for a polynomial to fit data

Test set error

$\min_{\Theta} J(\Theta)$

- $d = 1$ $\quad h_\theta(x) = \theta_0 + \theta_1 x \rightarrow \Theta^{(1)} \rightarrow J_{test}(\Theta^{(1)})$
- $d = 2$ $\quad h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \Theta^{(2)} \rightarrow J_{test}(\Theta^{(2)})$
- $d = 3$ $\quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 \rightarrow \Theta^{(3)} \rightarrow J_{test}(\Theta^{(3)})$
- $\vdots$ $\quad \vdots \quad \vdots$
- $d = 10$ $\quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \rightarrow \Theta^{(10)} \rightarrow J_{test}(\Theta^{(10)})$

  - Suppose $J_{test}(\Theta^{(5)})$ is the lowest among test errors
    - i.e. choose $\theta_0 + \theta_1 x + \cdots + \theta_5 x^5$

Model Selection

- Suppose $J_{test}(\Theta^{(5)})$ is the smallest among test errors
  - i.e. choose $\theta_0 + \theta_1 x + \cdots + \theta_5 x^5$
  - How well does the model generalize?
  - Problem
    - $J_{test}(\Theta^{(5)})$ is likely to be an optimistic estimate of generalization error.
      - i.e. *our extra parameter (d = degree of polynomial) is fit to test set*.
        - Chose it because the corresponding test set error is the smallest

Improved Model Selection

- Given a training set instead split into three pieces
  - Training set (60%) : m
  - Cross validation (CV) set (20%) : m_{cv}
  - Test set (20%) : m_{test}

- Calculate
  - Training error
    - $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
  - Cross validation error
    - $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$
  - Test error
    - $J_{test}(\theta) = \frac{1}{2m_{test}}\sum_{i=1}^{m_{test}}\left(h_\theta\left(x_{test}^{(i)}\right) - y_{test}^{(i)}\right)^2$

Improved Model Selection

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

Training set 60%

$$\left(x^{(1)}, y^{(1)}\right)$$
$$\left(x^{(2)}, y^{(2)}\right)$$
$$\vdots$$
$$\left(x^{(m)}, y^{(m)}\right)$$

Cross validation set 20%

$$\left(x_{cv}^{(1)}, y_{cv}^{(1)}\right)$$
$$\left(x_{cv}^{(2)}, y_{cv}^{(2)}\right)$$
$$\vdots$$
$$\left(x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})}\right)$$

Test set 20%

$$\left(x_{test}^{(1)}, y_{test}^{(1)}\right)$$
$$\left(x_{test}^{(2)}, y_{test}^{(2)}\right)$$
$$\vdots$$
$$\left(x_{test}^{(m_{test})}, y_{test}^{(m_{test})}\right)$$

Improved Model Selection

$d = 1 \quad h_\theta(x) = \theta_0 + \theta_1 x$

$d = 2 \quad h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$

$d = 3 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$

$\vdots$

$d = 10 \quad h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$

- Minimizing training error $J_{train}(\theta)$ for training set and then calculate each cross validation error
  - $\min_{\Theta} J_{train}(\Theta)$ by $(\theta_0 + \theta_1 x)$ ➔ $\theta^{(1)}$ ➔ $J_{cv}(\theta^{(1)})$
  - $\min_{\Theta} J_{train}(\Theta)$ by $(\theta_0 + \theta_1 x + \theta_2 x^2)$ ➔ $\theta^{(2)}$ ➔ $J_{cv}(\theta^{(2)})$

    …
  - $\min_{\Theta} J_{train}(\Theta)$ by $(\theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10})$ ➔ $\theta^{(10)}$ ➔ $J_{cv}(\theta^{(10)})$
- Pick the hypothesis with the lowest cross validation error.
- Estimate generalization error of model using the test set.

Improved Model Selection

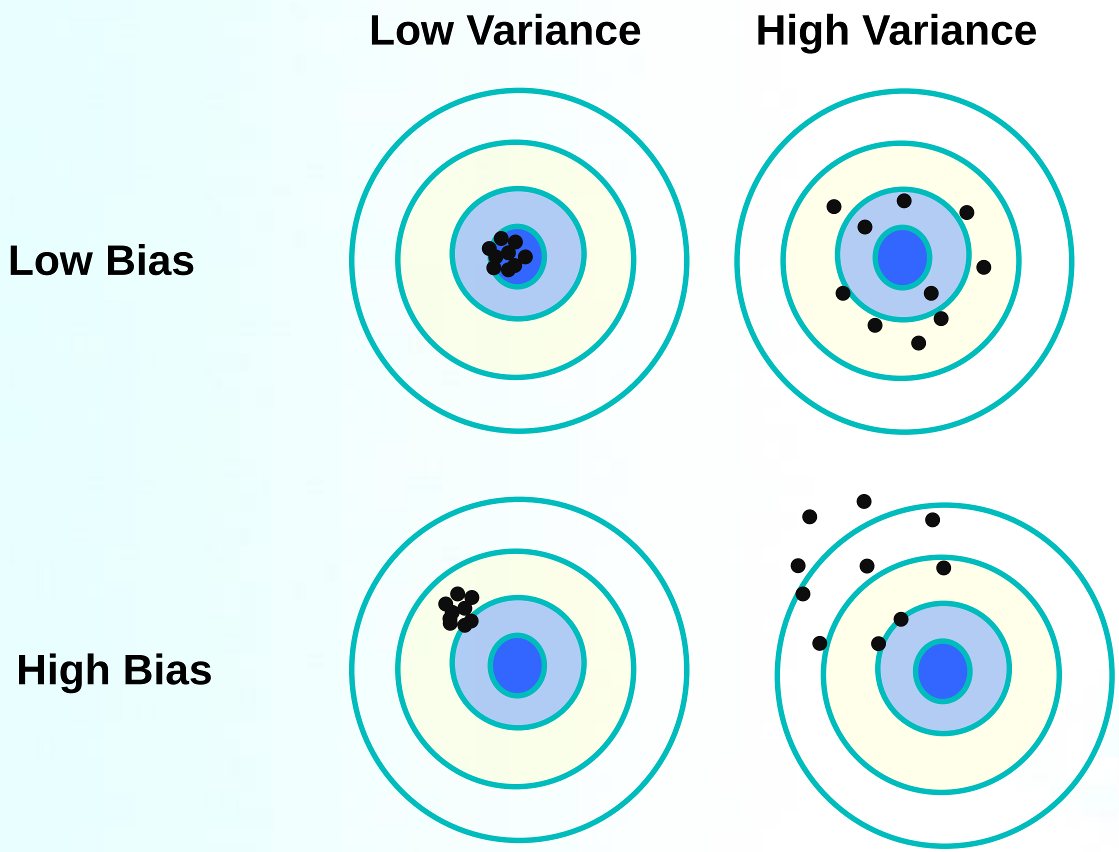
- Some people will still select the model using the test set
  - Then check the model is OK for generalization using the test error
    - With a MASSIVE test set, this is maybe OK

- But, making training and validation sets be separate
  - Much better

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Bias and Variance



Bias and Variance

- Bias
  - Error from erroneous assumptions in the learning algorithm
    - High bias can cause underfitting
      - Missing the relevant relations btw features and target outputs

- Variance
  - Error from sensitivity to small fluctuations in the training set
    - High variance can cause overfitting
      - Modeling the random noise in the training data, rather than the intended outputs.

Bias and Variance

- Error due to bias
  - Difference btw

    the expected (or average) prediction of our model and

    the correct value which we are trying to predict

- Error due to variance
  - Variability of a model prediction for a given data point

Bias and Variance

- Ideally, one wants to choose a model that
  - both accurately captures the regularities in its training data
    and generalizes well to unseen data.

  ➔ Unfortunately, it is typically impossible to do both simultaneously.

Bias and Variance

- Ideally, one wants to choose a model that
  - both accurately captures the regularities in its training data and generalizes well to unseen data.

  ➔ Unfortunately, it is typically impossible to do both simultaneously.

  - High-variance learning methods
    - May be able to represent their training set well, but are at risk of overfitting to noisy or unrepresentative training data
  - High bias ones
    - Typically produce simpler models that do not tend to overfit, but may underfit their training data
      - failing to capture important regularities.

Bias and Variance

- Assume that there is a function with noise

$$y = f(x) + \epsilon$$

where the noise, ϵ, has zero mean and variance σ^2

- Given a training set $(x_1,\ y_1)$, $(x_2,\ y_2)$, … ,$(x_m,\ y_m)$ from the above,
  - Find a function $\tilde{f}(x)$ that approximate the true function $f(x)$

    i.e. minimizing $E\left[\left(y - \tilde{f}(x)\right)^2\right]$ both for $x_1, x_2, \ldots, x_m$ and for future samples

  - $E\left[\left(y - \tilde{f}(x)\right)^2\right] = Bias\left[\tilde{f}(x)\right]^2 + Var\left[\tilde{f}(x)\right] + \sigma^2$
    - $Bias\left[\tilde{f}(x)\right] = E\left[\tilde{f}(x) - f(x)\right]$
    - $Var\left[\tilde{f}(x)\right] = E\left[\left(\tilde{f}(x) - E\left[\tilde{f}(x)\right]\right)^2\right] = E\left[\tilde{f}(x)^2\right] - E\left[\tilde{f}(x)\right]^2$

Bias and Variance

- Derivation
  - For one random variable X,
    - $Var[X] = E[(X - E[X])^2]$

      $= E[X^2 - 2XE[X] + E[X]^2]$

      $= E[X^2] - 2E[X]E[X] + E[X]^2$

      $= E[X^2] - E[X]^2$

      ➔ $E[X^2] = Var[X] + E[X]^2$

  - $E[y] = E[f + \epsilon] = E[f] + E[\epsilon] = f + 0 = f$ ($\because f$ is deterministic)
    - $Var[y] = E[(y - E[y])^2] = E[(y - f)^2] = E[(f + \epsilon - f)^2] = E[\epsilon^2] = \sigma^2$

Bias and Variance

- Derivation
  - $E\left[\left(y-\tilde{f}(x)\right)^2\right] = E\left[y^2+\tilde{f}^2-2y\tilde{f}\right] = E\left[y^2\right]+E[\tilde{f}^2]-E[2y\tilde{f}]$

$$= Var[y] + E[y]^2 + Var\left[\tilde{f}\right] + E[\tilde{f}]^2 - 2E[y]E\left[\tilde{f}\right]$$

$$= Var[y] + E[y]^2 + Var\left[\tilde{f}\right] + E[\tilde{f}]^2 - 2fE\left[\tilde{f}\right]$$

$$= Var[y] + f^2 + Var\left[\tilde{f}\right] + E[\tilde{f}]^2 - 2fE\left[\tilde{f}\right]$$

$$= Var[y] + Var\left[\tilde{f}\right] + f^2 - 2fE\left[\tilde{f}\right] + E\left[\tilde{f}\right]^2$$

$$= Var[y] + Var\left[\tilde{f}\right] + E[f-\tilde{f}]^2$$

$$= \sigma^2 + Var\left[\tilde{f}\right] + Bias[\tilde{f}]^2$$

Bias and Variance

- $E\left[\left(y-\tilde{f}(x)\right)^2\right] = Bias\left[\tilde{f}(x)\right]^2 + Var\left[\tilde{f}(x)\right] + \sigma^2$
  - $Bias\left[\tilde{f}(x)\right] = E\left[\tilde{f}(x) - f(x)\right]$
  - $Var\left[\tilde{f}(x)\right] = E\left[\left(\tilde{f}(x) - E\left[\tilde{f}(x)\right]\right)^2\right] = E\left[\tilde{f}(x)^2\right] - E\left[\tilde{f}(x)\right]^2$

- $Bias\left[\tilde{f}(x)\right]^2$
  - Error caused by the simplifying assumptions built into the method.

- $Var\left[\tilde{f}(x)\right]$
  - How much the learning method $\tilde{f}(x)$ will move around its mean

- σ^2
  - Irreducible error

Bias and Variance

- $E\left[\left(y-\tilde{f}(x)\right)^2\right] = Bias\left[\tilde{f}(x)\right]^2 + Var\left[\tilde{f}(x)\right] + \sigma^2$

- The more complex the model $\tilde{f}(x)$ is,
  the more data points it will capture,

➔ The lower the bias will be.

- However, complexity will make the model "move" more to capture the data points

➔ Hence its variance will be larger.

Bias and Variance

- Intuition
  - We should minimize bias even at the expense of variance
    - Presence of bias indicates something basically wrong with our model
  - A model with high variance could at least predict well on average,
    - At least it is not *fundamentally wrong*.

Bias and Variance

- Intuition
  - We should minimize bias even at the expense of variance
    - Presence of bias indicates something basically wrong with our model
  - A model with high variance could at least predict well on average,
    - At least it is not *fundamentally wrong*.

➔ This is mistaken logic.

Bias and Variance

- Intuition
  - We should minimize bias even at the expense of variance
    - Presence of bias indicates something basically wrong with our model
  - A model with high variance could at least predict well on average,
    - At least it is not *fundamentally wrong*.

➔ This is mistaken logic.

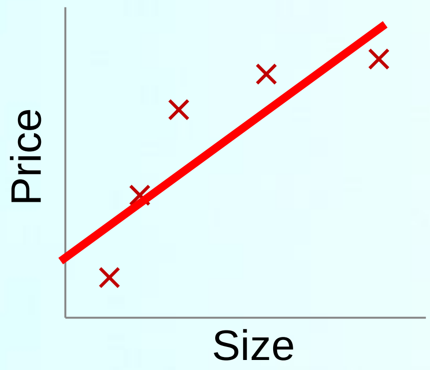
- It is correct that *a high variance and low bias model* can preform well in some sort of long-run average sense.
  - However, in practice modelers are always dealing with a single realization of the data set.
    - In these cases, long run averages are irrelevant
      - What is important is the performance of the model on the data we actually have
      - and in this case bias and variance are equally important
        - One should not be improved at an excessive expense to the other.

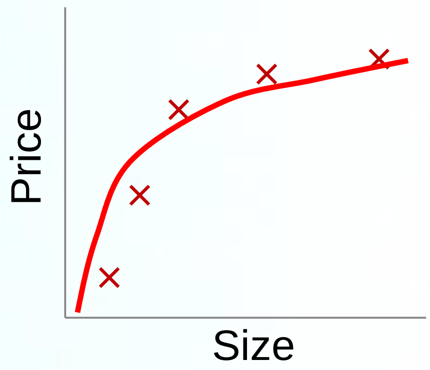
Outline

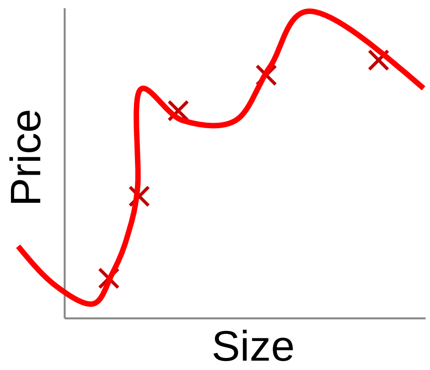
- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Bias and Variance

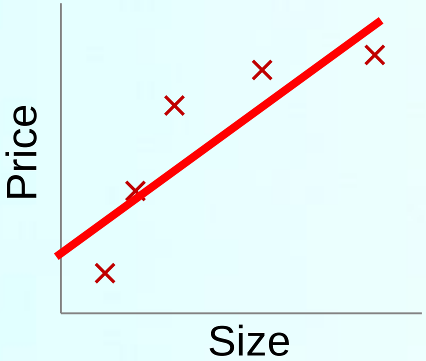
Bias and Variance





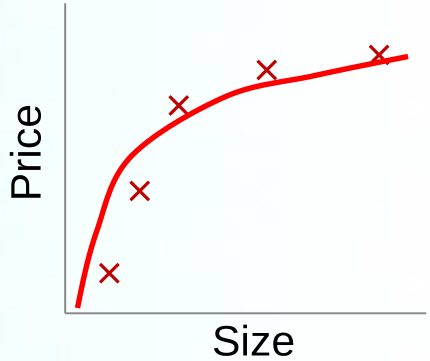


Bias and Variance



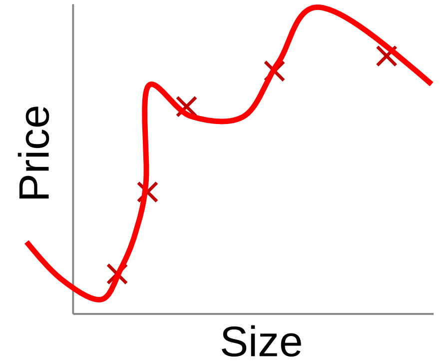
$\theta_0 + \theta_1 x$

High bias
(underfit)



$\theta_0 + \theta_1 x + \theta_2 x^2$

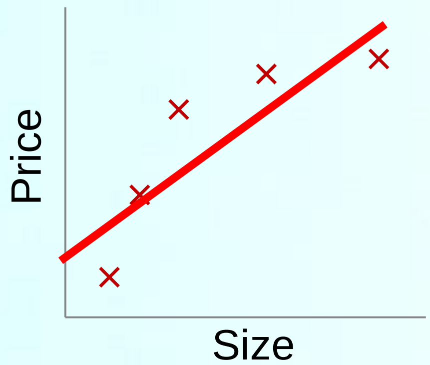
“Just right”



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

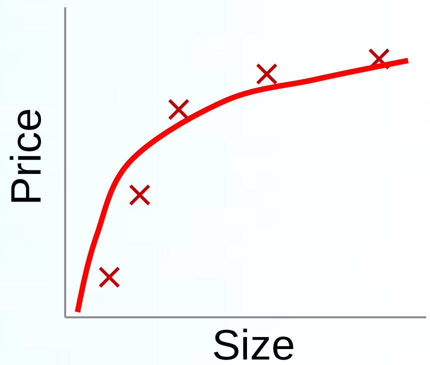
High variance
(overfit)

Bias and Variance



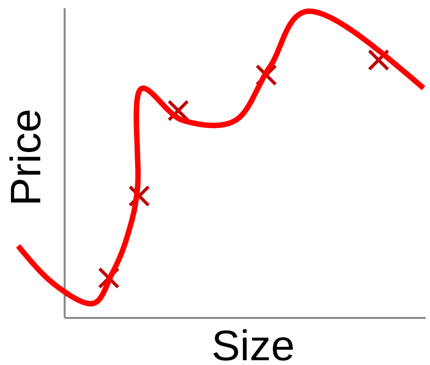
$\theta_0 + \theta_1 x$

High bias
(underfit)



$\theta_0 + \theta_1 x + \theta_2 x^2$

“Just right”



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

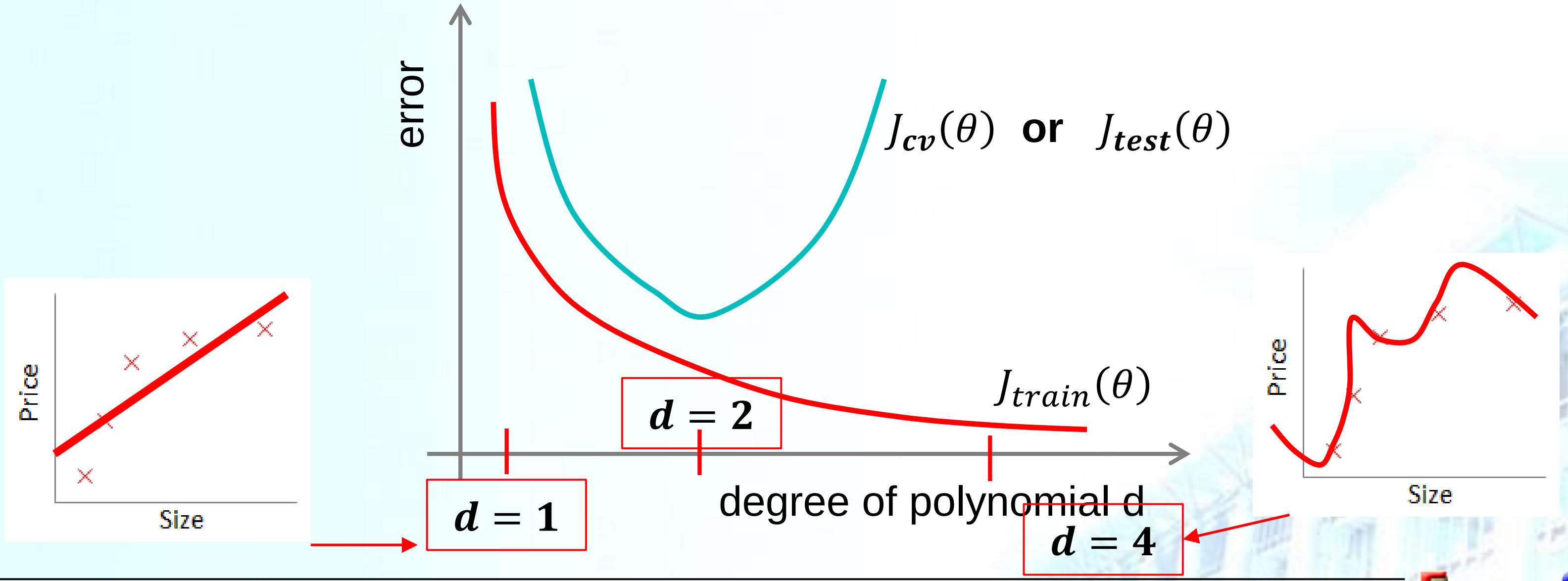
High variance
(overfit)

- High bias
  - Under fitting problem
- High variance
  - Over fitting problem

Bias and Variance

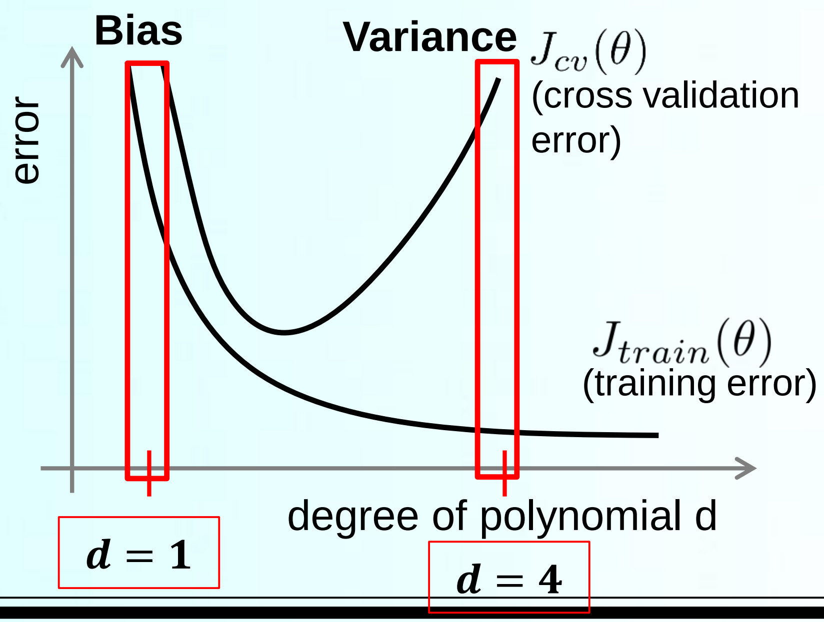
- The degree of a model will increase as we move towards overfitting

- Plot
  - x axis: degree of polynomial d
  - y axis: errors for both training and cross validation (two lines)
    - CV error and test set error will be very similar
  - ➔ $\boldsymbol{d = 2}$ can minimize both errors



Diagnosis of Bias and Variance

- If cv error is high
  - either (at the high end of $\boldsymbol{d}$) or (at the low end of $\boldsymbol{d}$)
    - if $\boldsymbol{d}$ is too small: this probably corresponds to a high bias problem
      - Underfit ➔ neither fit training data nor generalize
      - $J_{\boldsymbol{train}}(\theta)$ will be high, $J_{\boldsymbol{test}}(\theta) \approx J_{\boldsymbol{cv}}(\theta)$
    - if $\boldsymbol{d}$ is too large: this probably corresponds to a high variance problem
      - Overfit ➔ training set fits well but generalizes poorly
      - $J_{\boldsymbol{train}}(\theta)$ will be low, $J_{cv}(\theta) \gg J_{train}(\theta)$

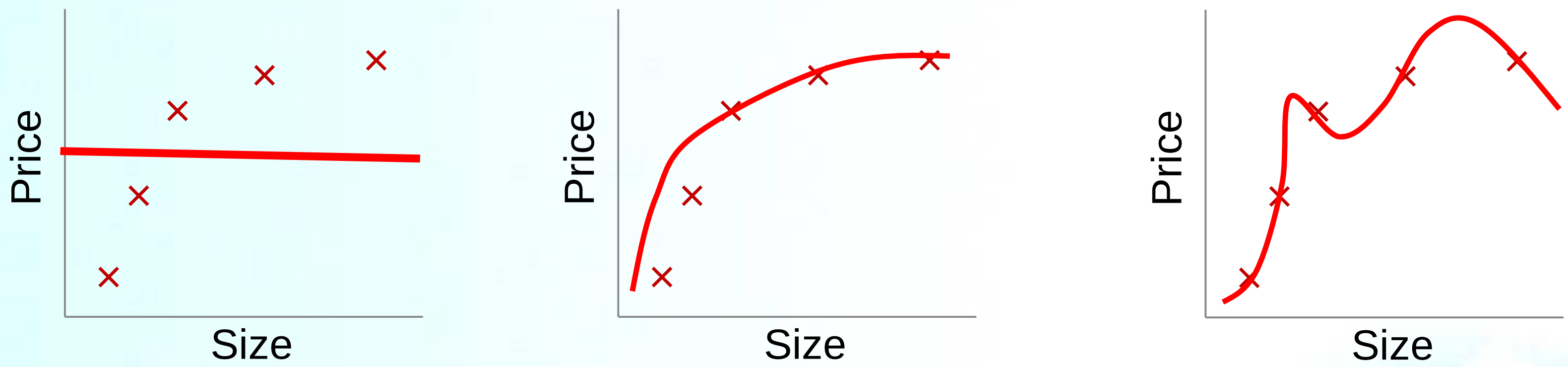


Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Linear Regression with Regularization

- Model
  - $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$
  - $J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda \sum_{j=1}^{n} \theta_j^2\right]$



Large λ
High bias (underfit)

$\lambda = 10000.\ \theta_1 \approx 0, \theta_2 \approx 0, \ldots$

$h_\theta(x) \approx \theta_0$

Intermediate λ
"Just right"

Small λ
High variance (overfit)

Choosing The Regularization Parameter λ

- Model
  - $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$
  - $J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda \sum_{j=1}^{n} \theta_j^2\right]$

- Define (without regularization term)
  - $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
  - $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$
  - $J_{test}(\theta) = \frac{1}{2m_{test}}\sum_{i=1}^{m_{test}}\left(h_\theta\left(x_{test}^{(i)}\right) - y_{test}^{(i)}\right)^2$

Choosing The Regularization Parameter λ

- Have a set or range of values to use

- Often increment by factors of 2 so
  - model(1)= λ = 0 ➔ $min_\theta J(\theta)$ ➔ $\theta^{(1)}$ ➔ calculate $J_{cv}(\theta^{(1)})$
  - model(2)= λ = 0.01 ➔ $min_\theta J(\theta)$ ➔ $\theta^{(2)}$ ➔ calculate $J_{cv}(\theta^{(2)})$
  - model(3)= λ = 0.02 ➔ $min_\theta J(\theta)$ ➔ $\theta^{(3)}$ ➔ calculate $J_{cv}(\theta^{(3)})$
  - model(4) = λ = 0.04
  - model(5) = λ = 0.08.

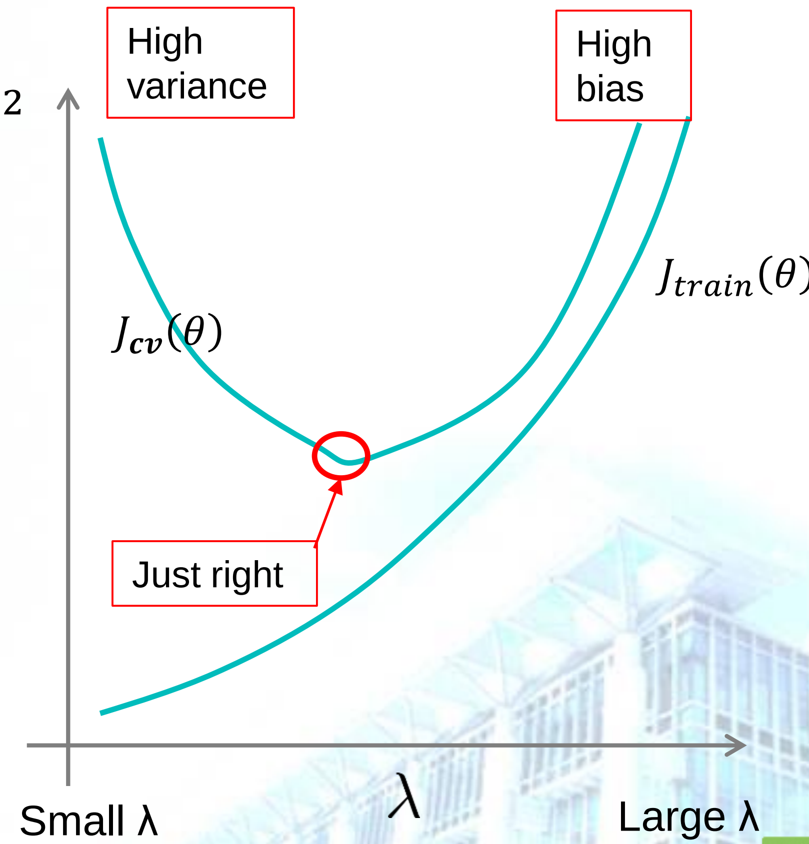
  …
  - model(12) = λ = 10.24 ➔ $min_\theta J(\theta)$ ➔ $\theta^{(12)}$ ➔ calculate $J_{cv}(\theta^{(12)})$

- Suppose $J_{cv}(\theta^{(5)})$ = min $\left(J_{cv}(\theta^{(1)}), \ldots, J_{cv}(\theta^{(12)}) \right)$
  - Then, calculate $J_{test}(\theta^{(5)})$

Bias/Variance as A Function of λ

- $J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n}\theta_j^2\right]$

- $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
- $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$



Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

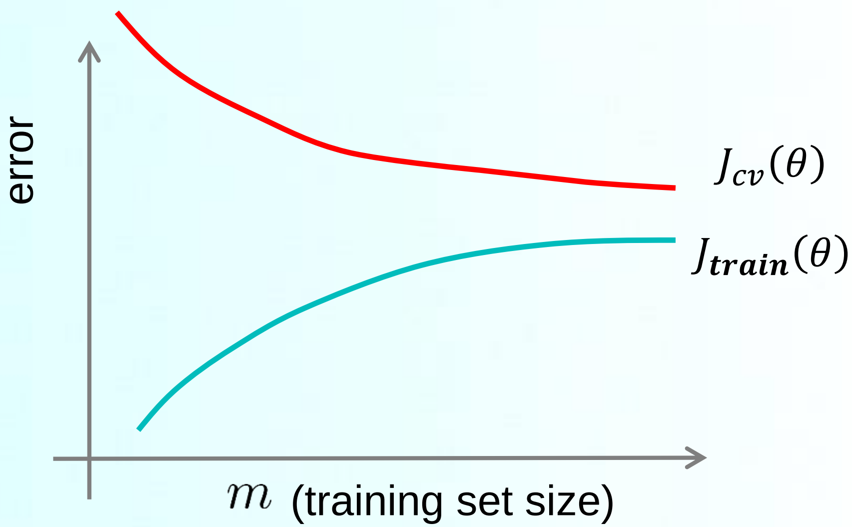
Learning Curve

- A learning curve
  - A graphical representation of the increase of learning (vertical axis) with experience (horizontal axis)

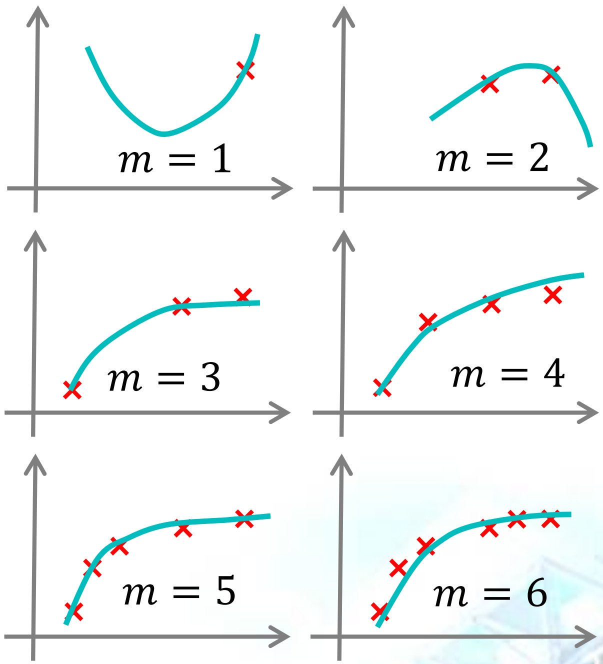
  - Plot J_{train} (average squared error on training set)
    or J_{cv} (average squared error on cross validation set)
    against m (number of training examples)

Learning Curves

- $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
- $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$

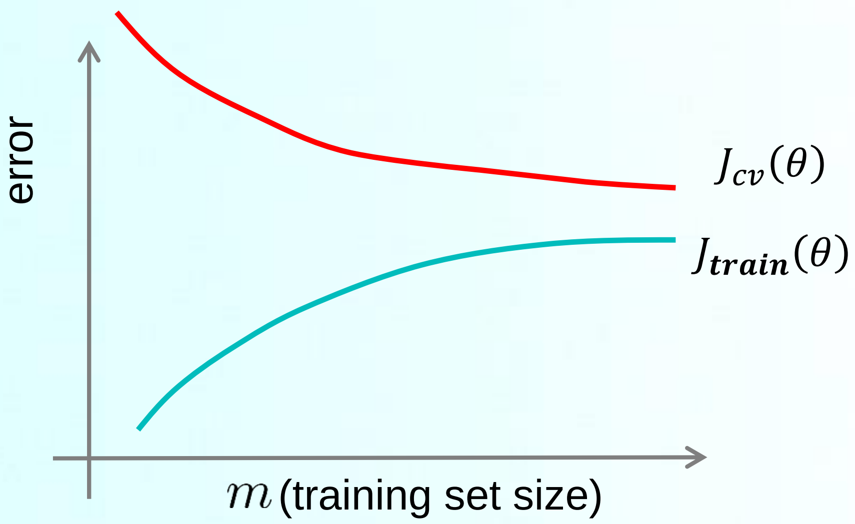


$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$



Learning Curves

- $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
- $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$



J_{train}

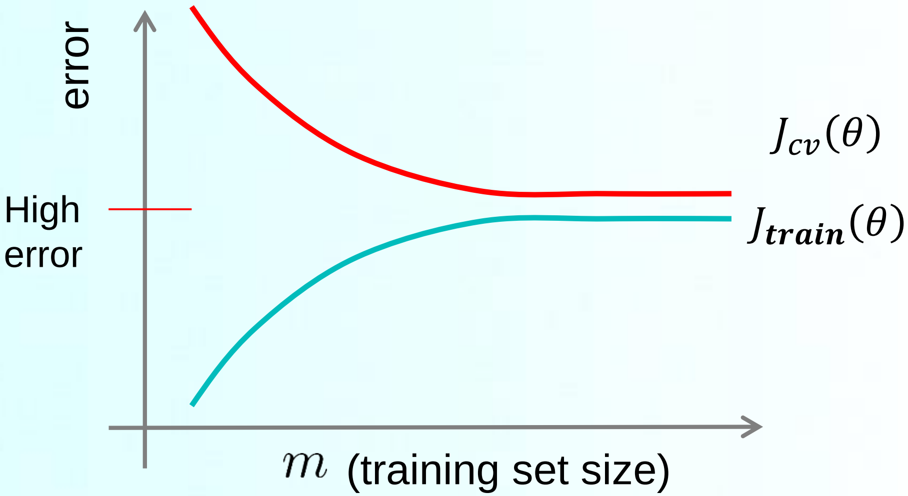
Error on smaller sample size is smaller
(as less variance to accommodate)
Error grows as m grows

J_{cv}

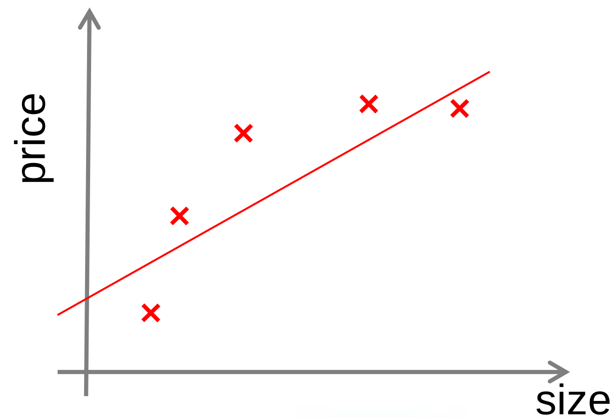
A tiny training set ➔ generalize badly
As m grows, our hypothesis generalize better
So, cv error will decrease as m grows.

High Bias

- If a learning algorithm is suffering from high bias,
  - (e.g. setting straight line to data)
  - getting more training data will not (by itself) help much.

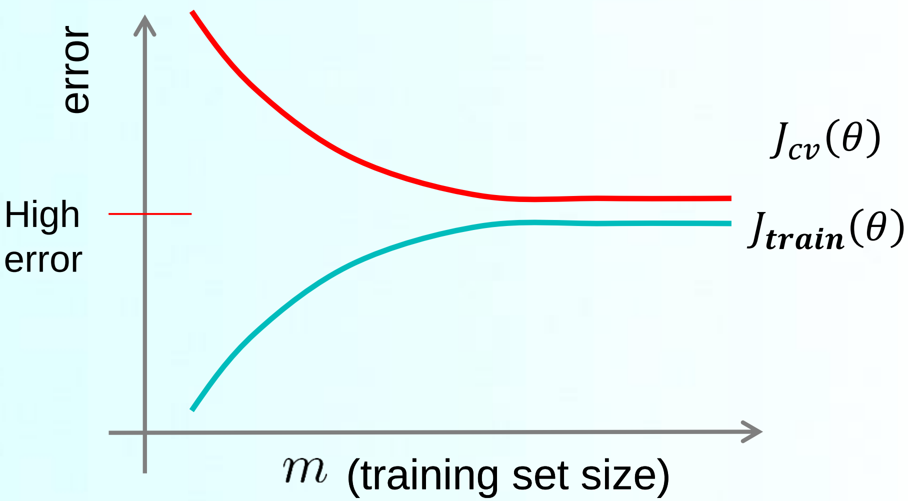


$$h_\theta(x) = \theta_0 + \theta_1 x$$



High Bias

- If a learning algorithm is suffering from high bias,
  - getting more training data will not (by itself) help much.



J_{train}
Error is small at first and grows
Error becomes close to cross validation
- So the performance of the cross validation and training set end up being similar (but very poor)

J_{cv}
Straight line fit is similar for a few vs. a lot of data
- So it does NOT generalize any better with lots of data because the function just does not fit the data
- No increase in data will help it fit

High Bias

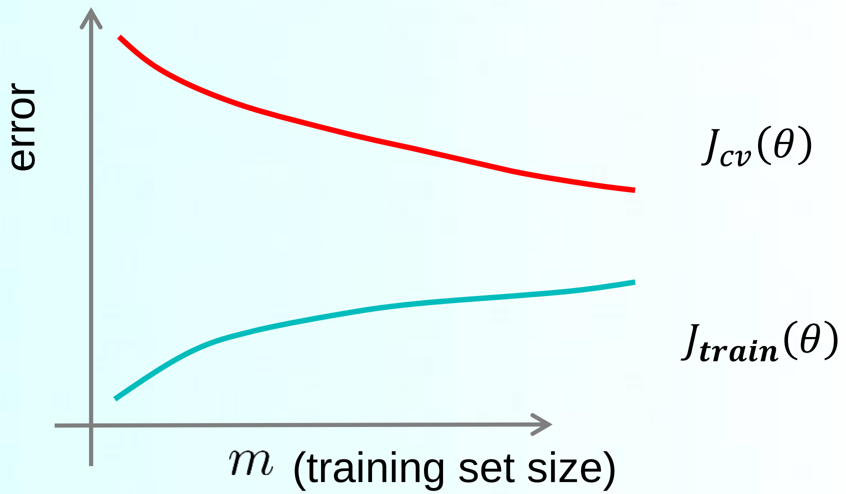
- If a learning algorithm is suffering from high bias,
  - getting more training data will not (by itself) help much.

- Cross validation and training errors in high bias
  - *both high*

- High bias
  - A problem with the underlying way we are modeling our data
    - So more data will not improve that model
      - It is too simplistic

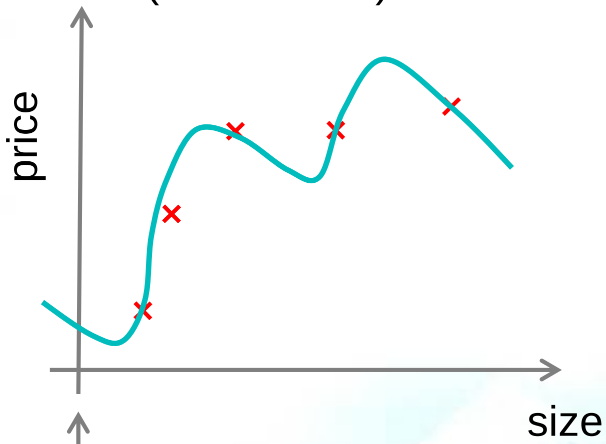
High Variance

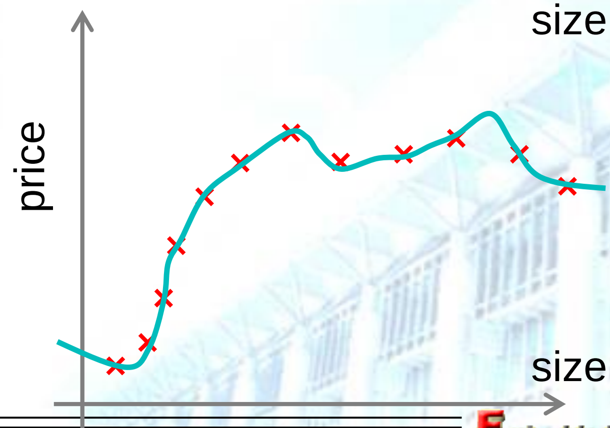
- If a learning algorithm is suffering from high variance,
  - e.g. high order polynomial
  - getting more training data is likely to help.



$$h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{100} x^{100}$$

(and small λ)

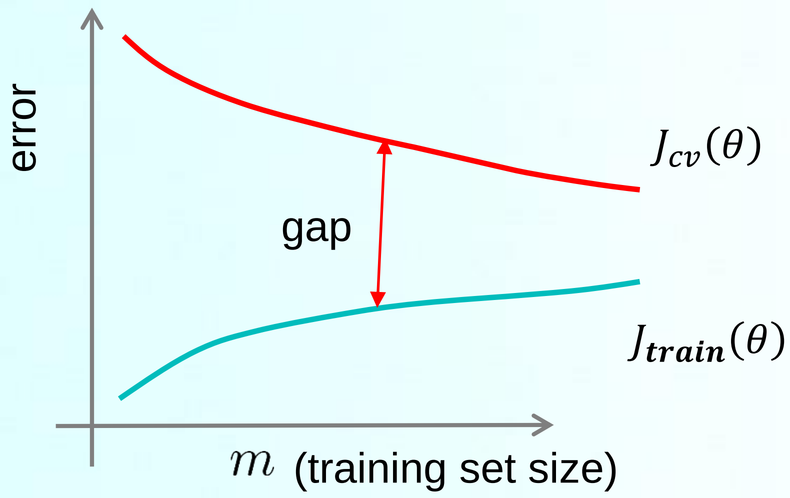




These are clean curves
- In reality, the curves we get are far dirtier
- learning curve plotting can help diagnose the problems our algorithm will be suffering from

High Variance

- If a learning algorithm is suffering from high variance,
  - getting more training data is likely to help.



J_{train}

When set is small, training error is small too

As training set sizes increases, value is still small

- But slowly increases (in a near linear fashion)
- Error is still low

J_{cv}

Error remains high,

- even when you have a moderate number of examples

The problem with high variance (overfitting)

- our model does NOT generalize

An indicative diagnostic to high variance

- A big gap btw training error and cross validation error

Outline

- Deciding what to try next - I
- Evaluating a hypothesis
- Model selection and training/validation/test sets
- Understanding of bias and variance
- Diagnosing bias vs. variance
- Regularization and bias/variance
- Learning curves
- Deciding what to try next - II

Debugging A Learning Algorithm

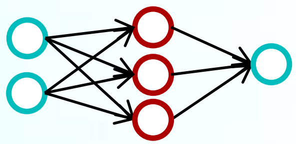
- One regularized linear regression to predict housing prices
  - However, when we test our hypothesis in a new set of houses, we find that it makes unacceptably large errors in its prediction.
    - What should we try next?

Debugging A Learning Algorithm

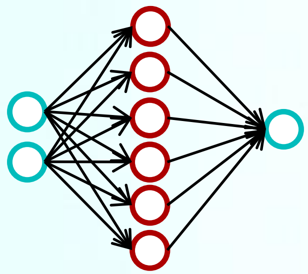
- Get more training examples ➔ helps to fix high variance
  - Not good if high bias
- Try smaller sets of features ➔ fixes high variance (overfitting)
  - Not good if high bias
- Try getting additional features ➔ fixes high bias (because hypothesis is too simple, make hypothesis more specific)
- Try adding polynomial features ➔ fixes high bias problem
- Try decreasing λ ➔ fixes high bias
- Try increasing λ ➔ fixes high variance

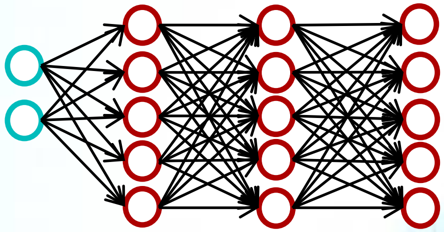
Neural Networks and Overfitting

- "Small" neural network
  - (fewer parameters; more prone to underfitting)
  - Computationally cheaper



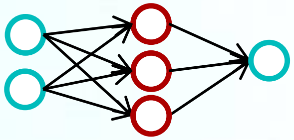
- "Large" neural network
  - (more parameters; more prone to overfitting)
    - Use regularization (λ) to address overfitting
  - Computationally more expensive

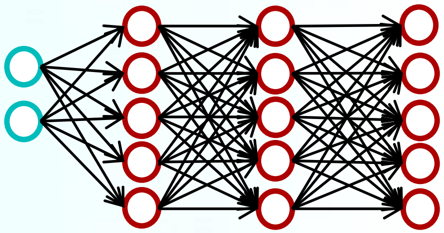




Neural Networks and Overfitting

- Using a single hidden layer is reasonable default
  - Try with 1, 2, 3 layers
    - See which performs best on cross validation set





References

- Andrew Ng, https://www.coursera.org/learn/machine-learning
- http://www.holehouse.org/mlclass/10_Advice_for_applying_machine_learning.html
- http://scott.fortmann-roe.com/docs/BiasVariance.html